

# Royale Heroes: A Unique RTS Game Using Deep Reinforcement Learning-based Autonomous Movement

1<sup>st</sup> Firdiansyah Ramadhan

*School of Computing*

*Telkom University*

Bandung, Indonesia

firdiar@student.telkomuniversity.ac.id

2<sup>nd</sup> Suyanto Suyanto

*School of Computing*

*Telkom University*

Bandung, Indonesia

suyanto@telkomuniversity.ac.id

**Abstract**—A game, a set of rules or policies by one or more people to carry out an action that will generate excitement for those who do it, can be developed using one of two types of playing methods: turn-based and real-time. In real-time strategic (RTS) games with many characters like Warcraft and Chess, it is fun to play. However, fun decreases since the players have to focus on many points in the playing area. Hence, autonomous movement (AM), an assistant to move the character, can be one solution. An AM can be implemented using deep reinforcement learning (DRL), a new method that can continue to learn without erasing previously learned memory that makes AM better since it can adapt to the playing style. This paper describes the Royale Heroes, a unique RTS game that implements a DRL-based AM. First, a base model is created using a rule-based model based on conventional artificial intelligence (AI). Next, a DRL model is trained to imitate the behavior of previously created AI. Evaluation of six matches shows that the DRL provides high accuracy of up to 98% to copy a behavior. Hence, it can be used as an AM to help players make choices in playing the game. When the player plays this game, the model will continue learning to imitate the player behavior.

**Keywords**—autonomous movement, deep reinforcement learning, games, real-time strategy, Royale Heroes

## I. INTRODUCTION

A game is a set of rules or policies by one or more people to carry out an action that will generate excitement for those who do it [1]. The game has two types of playing methods: Turn-Based and Real-Time [2], [3]. The Turn-Based is a way of playing where each player plays the game [4], where players must take turns and wait for their turn before playing again [5]–[7]. In contrast, Real-Time is one way of playing where the game is carried out continuously or simultaneously with other players, such as in [8]–[11].

There are several types of games that require players to focus on more than one point, such as real-time simulation games where the game runs directly without waiting and requires real-time resource management [12], [13]. In a strategy game that requires strategy design, resources, and individual management, players will find it challenging to make movements in the game [14], [15], [16].

Various techniques in machine learning, one of the branches of artificial intelligence (AI), are recently applied to the game area. Here, it plays a vital role in automatically making decisions and movements to run the game [17]. However, the use of AI implemented in the game has a predictable pattern to be deemed less effective. Hence, a unique AI is needed so that the game can run more interestingly.

ML can be a solution to the character movement patterns that can change. Machine Learning is a computer science that can learn according to the data it has. Deep reinforcement learning (DRL) is a new method combining deep learning and

reinforcement learning (RL). It produces an algorithm that can continue to learn without deleting or forgetting previous memory so that it can be useful as assistance for players because it can continue to learn and adapt to the game that is done by the player himself.

DRL is a type of unsupervised learning that does not have training data. Instead, DRL will classify data based on the rewards obtained from the actions taken. Since DRL is unsupervised learning, the search for training data is not required before conducting training, making the training process more flexible. It also allows for continuous training as long as the player plays the game.

Continuous training allows the DRL model to continue learning indefinitely, which means infinite copying of player behavior. However, of course, the player behavior is not static. A player who is a human will also learn and improvise to continue to give his best in battle. This process will continue without stopping. Players and the AM will continue to improvise and evaluate to be better than before. In other words, the AM will grow up together with the players.

Therefore, in this research, DRL is exploited to develop an AM for a unique RTS game called Royale Heroes. The literature review will be provided in Section 2. Next, Section 3 gives the Royale Heroes design and implementation. Section 4 discusses the experimental results. Section 5 gives the conclusion.

## II. LITERATURE REVIEW

In AI, there are many approaches to develop a game, such as AI planning [18], conventional machine learning (CML) [19], deep learning (DL) [19], reinforcement learning (RL) [20], and deep reinforcement learning (DRL) [21], [22]. CML enables a computer to learn from a given dataset for prediction, classification, and clustering [23]. Some studies on the implementation of CML in the games area have been performed, such as a simple real-time strategy game called Robot Defens, Checkers [24], Chess, Go, Shogi [20], Cricket [19]. Some CML methods commonly used in games are k-nearest neighbors (KNN), logistic regression (LR), Naive Bayes (NB), random forest (RF), and support vector machines (SVM), give high performances in games.

DL provides the advantages of features, which are generated automatically during the learning process, not manually created by humans as usually performed in developing a CML model [25], [26]. In [19], a feed-forward deep neural network (DNN) classifier has been proven to perform much higher than CML models: KNN, LR, NB, RF, and SVM. Convolutional neural network (CNN), one of the DL architectures, is proven to be able to recognize image patterns with an accuracy of up to 90% [27]. The convolutional layer can clarify patterns on existing input features to solve the problem of large input dimensions [28], [29]. The convolutional layer can be a way to speed up the